



go 语言编程





第三章

Part Three

- 1 选择结构
- 2 循环结构
- 3 跳转语句





第三章 流程控制



3.1 选择结构

选择结构是根据条件表达式的结果选择执行某些语句。在Go语言中选择结构主要通过两种语句实现，if语句和switch语句。

3.1.1 if语句的语法：

```
// 1
if condition {
    ...
}
// 2
if condition {
    ...
} else {
    ...
}
```

```
// 3
if condition {
    ...
} else if condition {
    ...
}
// 4
if condition {
    ...
} else if condition {
    ...
} else {
    ...
}
```

第三章 流程控制



if语句注意事项:

- 1.不需要使用括号()将条件包含起来。
- 2.无论语句体内有几条语句，花括号{}都必须存在。
- 3.左花括号{必须与if或else同处一行，否则有编译器错误。
- 4.else必须和上一个语句体结束的右花括号}同处一行，否则有编译错误。
- 5.在if之后可以添加变量初始化语句，但需要使用分号;间隔。

惯用法：

```
if v, err := func(); err == nil {  
    //逻辑语句  
}
```



第三章 流程控制



3.1.2 switch语句

switch语句的语法：

```
switch condition {  
    case v0:  
        ...  
    case v1:  
        ...  
    case v2:  
        ...  
    case v3:  
        ...  
    case v4, v5, v6:  
        ...  
    default:  
        ...  
}
```



第三章 流程控制



switch语句注意事项:

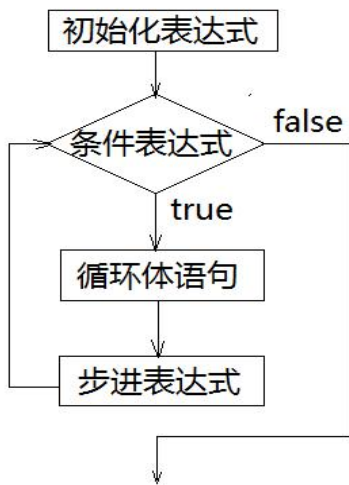
1. 左花括号{必须与switch同处一行。
2. case后可以使用常量, 还可以使用变量。
3. 单个case后可以出现多个结果选项。
4. 不需要用break来明确退出一个case。
5. 只有在case中明确添加fallthrough关键字, 才会继续执行下一个case。
6. 可不设定switch后的条件表达式, 在此种情况下整个switch结构与if...else...的逻辑等同。



3.2 循环结构

3.2.1 for语句的语法格式

```
// 普通循环  
for 初始化表达式; 条件表达式; 步进表达式 {  
    循环体语句  
}
```



for循环的程序流程图

第三章 流程控制



for语句注意事项:

- 1)只支持for结构，不支持while和do-while结构。
- 2)左花括号{必须与for处于同一行。
- 3)不支持以逗号为间隔的多赋值语句，支持使用多元赋值的方式来初始化多个变量。这个很好区分，在一个表达式中不允许有多个赋值语句，即:=或=。



多元赋值与多赋值语句的区别

```
array := []int{1, 2, 3, 4, 5, 6}
for i, j := 0, len(array)-1; i < j; i, j = i+1, j-1 {
    array[i], array[j] = array[j], array[i]
}
fmt.Println(array)
```

```
array := []int{1, 2, 3, 4, 5, 6}
for i:=0, j:=len(array)-1; i < j; i= i+1,j=j-1 {
    array[i], array[j] = array[j], array[i]
}
fmt.Println(array)
```

错误的

第三章 流程控制



3.2.2 continue和break语句

continue用于结束本次循环，而break语句结束整个循环。

3.3 goto语句

goto语句是跳转语句，一般要和标签配合使用。可以让程序直接跳转到标签之处继续执行。注意标签后加：号。

```
goto LABEL  
...  
...  
..  
LABEL:
```



T H A N K S

感谢聆听，期待反馈

